**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings of claims in the application:

**Listing of Claims:**

1. (Currently amended) A method of self-programming a graphics processing unit (GPU), the method comprising:

receiving from ~~the CPU~~ a CPU a blit instruction defining a blit operation, wherein a destination for the blit operation corresponds to a control register; and

storing a first control value in ~~a control register~~ the control register using the blit operation, wherein the first control value in the control register determines ~~the behavior~~ a behavior of the GPU.

2. (Original) The method of claim 1, further comprising applying the blit operation to a second control value to determine the first control value.

3. (Original) The method of claim 2, wherein the second control value is stored in a memory.

4. (Original) The method of claim 3, wherein the memory is a second control register.

5. (Original) The method of claim 3, wherein the second control value is stored in a table of control values accessed by an index value.

6. (Original) The method of claim 5, further comprising changing the index value to access a third control value in the table following the blit operation.

7. (Original) The method of claim 2, wherein the second control value is a starting memory address for a display buffer.

8. (Original) The method of claim 2, wherein the second control value is a clip plane distance.

9. (Original) The method of claim 8, wherein the second control value is greater than the depth extent of an object.

10. (Original) The method of claim 1, wherein the blit operation includes a colorkey operation.

11. (Original) The method of claim 1, wherein the blit operation includes a logic operation on the first control value.

12. (Original) The method of claim 1, wherein the blit operation includes a pattern copy operation.

13. (Original) The method of claim 2, wherein the first control value is a copy of the second control value.

14. (Original) The method of claim 1, wherein receiving comprises reading a blit instruction from a command buffer asynchronously with the CPU.

15. (Currently amended) A graphics rendering system comprising:

a central processing unit;

a graphics processing unit adapted to create graphics data in response to a graphics command, wherein the graphics processing unit has a control register and a blit engine; and

a command buffer adapted to receive at least a first graphics command from the central processing unit and further adapted to communicate the first graphics command to the graphics processing unit;

wherein, the blit engine is adapted to perform a blit operation that stores ~~store~~ a first control value in the control register in response to the first graphics command.

16. (Original) The graphics rendering system of claim 15, wherein the blit engine is adapted to apply a blit operation to a second control value to determine the first control value.

17. (Original) The graphics rendering system of claim 16, further comprising a memory adapted to store the second control value.

18. (Original) The graphics rendering system of claim 17, wherein the memory is a second control register.

19. (Original) The graphics rendering system of claim 17, wherein the memory comprises a table of control values including the second control value and accessed by an index value.

20. (Original) The graphics rendering system of 16, wherein the second control value is a starting memory address for a display buffer.

21. (Original) The graphics rendering system of claim 16, wherein the second control value is a clip plane distance.

22. (Original) The graphics rendering system of claim 16, wherein the blit engine is adapted to perform a colorkey operation on the second control value.

23. (Original) The graphics rendering system of claim 15, wherein the blit engine is adapted to perform a logic operation on the first control value.

24. (Currently amended) A method of self-programming a graphics processing unit (GPU) using a set of GPU commands created by a driver, the method comprising:

writing a first set of rendering commands to a command buffer;

writing a set of self-programming commands including a blit instruction to the command buffer, wherein a destination for the blit instruction corresponds to a control register of the GPU; and

writing a second set of rendering commands to the command buffer.

25. (Original) The method of claim 24, wherein the first set of rendering commands, the set of self-programming commands, and the second set of rendering commands are written sequentially to the command buffer.

26. (Original) The method of claim 24, wherein the first set of rendering commands is associated with a first image to be rendered to a first display buffer, the second set of rendering commands is associated with a second image to be rendered to a second display buffer, and the set of self-programming commands is adapted to instruct the GPU to display the first image.

27. (Original) The method of claim 26, wherein the blit instruction is adapted to store a memory address associated with the first display buffer in a control register of the GPU.

28. (Original) The method of claim 24, wherein the first set of rendering commands is associated with a bounding box associated with an object and the second set of rendering commands is associated with the object; and further wherein the set of self-programming commands is adapted to instruct the GPU to evaluate the visibility of the bounding box and to optionally bypass the rendering of the object in response to the evaluation of the visibility of the bounding box.

29. (Original) The method of claim 28, wherein the blit instruction includes a colorkey operation on a register of the GPU.

30. (Original) The method of claim 29, wherein the register of the GPU is a rendered pixel count register.

31. (Original) The method of claim 29, wherein the blit instruction is adapted to store a control value in a control register in response to the colorkey operation, and wherein the control value is adapted to instruct the GPU to abort the processing of the second set of rendering commands.

32. (Original) The method of claim 31, wherein the control value is a clip plane distance.

33. (Currently amended) An information storage medium having a set of instructions adapted to direct an information processing device in communication with a graphics processing unit (GPU) to perform the steps of:

writing a first set of rendering commands to a command buffer;

writing a set of self-programming commands including a blit instruction to the command buffer, wherein a destination of the blit instruction corresponds to a control register of the GPU; and

writing a second set of rendering commands to the command buffer.

34. (Original) The information storage medium of claim 33, wherein the first set of rendering commands, the set of self-programming commands, and the second set of rendering commands are written sequentially to the command buffer.

35. (Original) The information storage medium of claim 33, wherein the first set of rendering commands is associated with a first image to be rendered to a first display buffer, the second set of rendering commands is associated with a second image to be rendered to a second display buffer, and the set of self-programming commands is adapted to instruct the GPU to display the first image.

36. (Original) The information storage medium of claim 35, wherein the blit instruction is adapted to store a memory address associated with the first display buffer in a control register of the GPU.

37. (Original) The information storage medium of claim 33, wherein the first set of rendering commands is associated with a bounding box associated with an object and the second set of rendering commands is associated with the object; and further wherein the set of self-programming commands is adapted to instruct the GPU to evaluate the visibility of the bounding box and to optionally bypass the rendering of the object in response to the evaluation of the visibility of the bounding box.

38. (Original) The information storage medium of claim 37, wherein the blit instruction includes a colorkey operation on a register of the GPU.

39. (Original) The information storage medium of claim 38, wherein the register of the GPU is a rendered pixel count register.

40. (Original) The information storage medium of claim 38, wherein the blit instruction is adapted to store a control value in a control register in response to the colorkey operation, and wherein the control value is adapted to instruct the GPU to abort the processing of the second set of rendering commands.

41. (Original) The information storage medium of claim 40, wherein the control value is a clip plane distance.